Agosto 2014

Proyecto de Desarrollo de Aplicaciones Web

Juego Jeopardy

Actividad por equipos de 2 ó 3 personas

Objetivo

El objetivo del proyecto es construir una aplicación web que implemente una versión pedagógica del popular juego Jeopardy (http://es.wikipedia.org/wiki/Jeopardy!).

El sistema debe estar orientado a la pedagogía, las categorías deben ser temas de una clase, las pistas deben ser respuestas a preguntas relacionadas con la clase y las ganancias deben expresarse en puntos en vez de dinero. Los alumnos que concursan deben registrarse y el desarrollo del juego debe quedar grabado para acreditar las puntuaciones obtenidas por los alumnos que concursaron así como para obtener estadísticas generales del aprovechamiento de los alumnos.

Funcionalidad

Los profesores serán capaces de:

- Conectarse al sistema mediante un “userid” y un “password” válidos. Los profesores reciben por e-mail un password inicial, el cual deberán de cambiar la primera vez que accedan al sistema. Si se escribe 3 veces un password incorrecto, la cuenta se bloquea. Cada vez que el usuario teclea un password incorrecto, el sistema le previene que su cuenta está a punto de ser bloqueada.

- Buscar un tema en particular ya sea tecleando las primeras letras del tema o bien seleccionando entre una lista de temas desplegados. Al encontrarlo se indica al sistema para que lo agregue en un perfil (instancia del juego) que será utilizado en el salón de clase.
- Seleccionar las categorías y pistas de cada tema e incorporarlas al perfil.
- Modificar el perfil editando los temas y pistas seleccionados.
- Ejecutar el juego en el salón de clase. Los alumnos pueden participar individualmente o por equipos. El resultado del juego (alumnos participantes y puntos obtenidos) debe ser registrado en la base de datos.
- Obtener reportes de los resultados de todos los juegos efectuados durante el semestre.
- Crear clases (matemáticas, física, redes, …) y para cada clase crear categorías y pistas
- Agregar en cualquier momento nuevas categorías y pistas
- Editar las pistas existentes (borrar, cambiar el valor del premio, cambiar la redacción etc.)
- Editar las categorías existentes (borrar, cambiar la redacción, etc.)

Interfaz

La interfaz de usuario del programa debe tener un estilo consistente entre páginas.

El proyecto debe utilizar las siguientes tecnologías:

Del lado del cliente

- HTML
- CSS
- JavaScript
- Validación de formas
- Ajax

Del lado del servidor

El servidor debe instalarse en un servidor Tomcat stand alone, tal como se instalaría en el servidor de un cliente. No debe hacerse la demostración usando Netbeans ya que es una herramienta de desarrollo. Cada equipo deberá investigar cómo se realiza la transferencia del programa desarrollado en Netbeans hacia un servidor Tomcat este tema no se verá en clase.

- Servlets, JSP, PHP ó cualquier otra tecnología presentada en clase
- El modelo MVC
- Base de datos

Entregables

- **Entrega en Blackboard, 24 de noviembre de 2014.** Entregar en Blackboard los programas cliente (HTML, CSS, JavaScript) ; servidor (Servlet, JSP, Java Beans, PHP, …) y un dump de la base de datos (SQL). Entregar además un reporte describiendo la arquitectura del servidor incluyendo los clientes, los servidores, las clases y sus relaciones, y un reporte con el diseño de la base de datos; el manual de instalación y el manual de usuario. Solo se requiere subir el reporte a blackboard una sola vez. Cualquiera de los integrantes del equipo lo puede subir.
- **Demostración durante la sesión del 25 de noviembre de 2014**. Se hará una demostración del programa en el salón de clases..

Sin excepción, todos los integrantes del equipo deben estar presentes en la revisión.

Rúbrica para la entrega

Aspecto a revisar	**Penalización**
No usa JavaScript	40%
No usa CSS	30%
Páginas incompletas	Hasta 40%
No usa MVC	40%
No usa Ajax	20%
No usa Base de Datos	40%
No incluye e-mail	10%
No incluye diagrama con la arquitectura	10%
Documentación incompleta	Hasta 30%
Funcionalidad	Hasta 100%
Presentación	Hasta 10%
Estilo de codificación *	Hasta 10%

* Estilo de codificación. Para que el código sea legible, fácil de mantener y de reutilizar, se deben seguir buenas prácticas de programación como por ejemplo:

- Uso adecuado de comentarios.
- Uso adecuado de espacios en blanco.
- Uso adecuado de indentación.
- Nombres adecuados para todas las clases, métodos y variables. Deben relacionarse con su uso.
- No incluir código muerto (instrucciones, variables, métodos no utilizados).